



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/741,516	12/19/2003	Thomas O'Rourke	920673-95339	4435

23644	7590	10/31/2007
BARNES & THORNBURG LLP		
P.O. BOX 2786		
CHICAGO, IL 60690-2786		

EXAMINER	
HICKS, MICHAEL J	

ART UNIT	PAPER NUMBER
2165	

NOTIFICATION DATE	DELIVERY MODE
10/31/2007	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

patent-ch@btlaw.com

D

Office Action Summary	Application No. 10/741,516	Applicant(s) O'ROURKE ET AL.	
	Examiner Michael J. Hicks	Art Unit 2165	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 10 August 2007.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-24 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-24 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 09 December 2003 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
 2. ☐ Certified copies of the priority documents have been received in Application No. _____.
 3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

[Signature]
Merren Abel-Jail
Primary Examiner

Attachment(s)

- | | |
|--|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | 5) <input type="checkbox"/> Notice of Informal Patent Application |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____ |

DETAILED ACTION

1. Claims 1-24 Pending.

Response to Arguments

2. Applicant's arguments, see response, filed 8/10/2007, with respect to the rejection(s) of claim(s) 1-23 under USC 102 and USC 103 have been fully considered and are persuasive. Therefore, the rejection has been withdrawn. However, upon further consideration, a new ground(s) of rejection is made in view of the new art of Doshi et al. (U.S. Patent Number 6,144,667 and referred to hereinafter as Doshi) in combination with the previously cited art of Stal.

Claim Rejections - 35 USC § 102

3. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

4. Claim 24 rejected under 35 U.S.C. 102(b) as being anticipated by Doshi.

As per Claim 24, Doshi discloses a method of providing an asynchronous interaction between a client and a server (i.e. *Column 6, Lines 12-20 which disclose that the connection I maintained over a standard internet connection.*), comprising the steps of providing Web Services on the server to be consumed by the client (i.e. *Column 3, Lines 20-48, wherein*

Art Unit: 2165

the network gateway acts in response to commands by the client in order to control the application.) and providing Web Services on the client to be consumed by the server (i.e. Column 3, Lines 20-48, wherein the network gateway sends responses and status updates to the client. Also note Column 9, Lines 60-64 which indicate that notifications may be sent to the client which are independent of application control commands sent by the client (e.g. third party bridge notifications).), whereby each of the client and server acts as both a Web Services provider and a Web Services consumer, such that when acting as a consumer each can notify the other of events asynchronously by invoking a Web Services command (i.e. Column 3, Lines 20-48, wherein the network gateway sends responses and status updates to the client. Also note Column 9, Lines 60-64 which indicate that notifications may be sent to the client which are independent of application control commands sent by the client (e.g. third party bridge notifications). Note that the client sending requests may be considered to be event notification messages.), and wherein said asynchronous interaction is provided without maintaining a dedicated communications channel (i.e. *Column 6, Lines 12-20 which disclose that the connection I maintained over a standard internet connection.*) (Page 71, Column 1, Paragraph 1; Page 74, Column 2, Paragraph 4), and wherein the server notifies the client of events occurring independently of any notification received by the server from the client (i.e. Column 3, Lines 20-48, wherein the network gateway sends responses and status updates to the client. Also note Column 9, Lines 60-64 which indicate that notifications may be sent to the client which are independent of application control commands sent by the client (e.g. third party bridge notifications).).

Claim Rejections - 35 USC § 103

5. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

6. Claims 1-23 rejected under 35 U.S.C. 103(a) as being unpatentable over Doshi in view of Stal.

As per Claim 1, 11, and 12, Doshi discloses a computer program product, server (e.g. system), and method of providing a client computer with remote access to an application controlled by a server across a data network (i.e. Figure 1a) without maintaining a dedicated communications channel between the client and the server (i.e. *Column 6, Lines 12-20 which disclose that the connection I maintained over a standard internet connection.*), comprising the steps of: a) providing a network addressable server-side service which executes one or more predefined procedures to control said application in response to predefined application control commands received by the server over the network (i.e. *Column 3, Lines 20-48, wherein the network gateway acts as the server to control the application (e.g. call session and voice switch).*); c) providing instructions accessible over the data network by the client which define a network addressable client-side service which executes one or more predefined procedures to generate notifications on the client in response to predefined notification commands received by the client over the network (i.e. *Column 3, Lines 20-48, wherein the network gateway sends responses and status updates to the*

client. Also note Column 9, Lines 60-64 which indicate that notifications may be sent to the client which are independent of application control commands sent by the client (e.g. third party bridge notifications.); and d) providing on the server an interface between the application and the data network which is effective to issue one or more of said notification generation commands in response to notifications of events received from the application, said events occurring and said notification generation commands issuing independently of said application control commands received from the client (i.e. Column 3, Lines 20-48, wherein the network gateway sends responses and status updates to the client. Also note Column 9, Lines 60-64 which indicate that notifications may be sent to the client which are independent of application control commands sent by the client (e.g. third party bridge notifications.); whereby the server-side service provides a network addressable control service to enable the client to control said application on the server (i.e. Column 3, Lines 20-48, wherein the network gateway acts in response to commands by the client in order to control the application.), and the client-side service provides a network addressable notification service to enable the server to notify the client of events occurring in the application independently of said application control commands received from the client (i.e. Column 3, Lines 20-48, wherein the network gateway sends responses and status updates to the client. Also note Column 9, Lines 60-64 which indicate that notifications may be sent to the client which are independent of application control commands sent by the client (e.g. third party bridge notifications).)

Doshi fails to disclose providing a definition of said application control commands, said definition being accessible over the data network by a client to enable the client to provide an interface which formulates said predefined commands and addresses said commands to the server-side service in response to inputs to the client.

Stal discloses providing a definition of said application control commands, said definition being accessible over the data network by a client to enable the client to provide an interface which formulates said predefined commands and addresses said commands to the server-side service in response to inputs to the client (i.e. "Step 2: *Implementing an Interface Definition Language. In the next implementation step, an interface definition language is introduced to express structural and deployment information in an implementation- neutral fashion. From the specifications written in this language, tools will automatically generate the implementations of client-side and serverside proxies. Again, XML represents the appropriate means to define such a data representation language. As a result WSDL (Web Services Description Language)—see www.w3c.org/TR/wsd1—was created providing the following constituents: • Types are used as core elements to build messages (XML Schema Notation). • Messages define packages exchanged within a single message transfer. Requests and responses represent separate messages. • Porttypes group messages to abstract operations. • Bindings map Porttypes to concrete protocols. • Ports denote the concrete communication addresses of services. • A Service comprises a collection of ports. Step 3: *Implementing a Service Directory. Before a client can access a service, it must find the service. For this purpose, a central broker must be available that allows implementers to register their services as well as clients to locate these services. Again, XML denotes the core technology to store and retrieve service registrations. UDDI (Universal Discovery, Description, and Integration)—see www3.ibm.com/services/uddi/standard.html—provides all functionality expected from a service broker. In UDDI, servers use the Publishers API to register services as well as additional business information with the global repository (see Figure 3). Clients access the Inquiry API to browse the repository and retrieve service descriptions. SOAP is used as communication protocol in all interactions. The client obtains the WSDL description from the UDDI repository both dynamically or statically, generates a client-side proxy, and invokes the Web service.*" The preceding text excerpt clearly indicates that a service definition language (WSDL) is used to create a description of the application control commands and the interface,*

including responses to user input. Also this definition is available to be sent over the network from the server to the client.) (Page 74, Column 1, Paragraph 3, Column 2, Paragraph 1).

It would have been obvious to one skilled in the art at the time of Applicants invention to modify the teachings of Doshi with the teachings of Stal to include providing a definition of said application control commands, said definition being accessible over the data network by a client to enable the client to provide an interface which formulates said predefined commands and addresses said commands to the server-side service in response to inputs to the client with the motivation of providing a standard web service definition using WSDL.

As per Claims 2 and 14, Doshi discloses said server-side service and said client-side service are each provided as web services between a provider and a remote consumer, the consumer of the server-side service being the client interface (i.e. Column 3, Lines 20-48, wherein the network gateway acts in response to commands by the client in order to control the application.) and the consumer of the client-side service being the server interface (i.e. Column 3, Lines 20-48, wherein the network gateway sends responses and status updates to the client. Also note Column 9, Lines 60-64 which indicate that notifications may be sent to the client which are independent of application control commands sent by the client (e.g. third party bridge notifications).).

As per Claims 3 and 15, Doshi fails to disclose discloses the definition of said application control commands is a web services description language (WSDL) file provided on the server.

Stal discloses the definition of said application control commands is a web services description language (WSDL) file provided on the server (i.e. *"Step 2: Implementing an Interface Definition Language. In the next implementation step, an interface definition language is introduced to express structural and deployment information in an implementation- neutral fashion. From the specifications written in this language, tools will automatically generate the implementations of client-side and serverside proxies. Again, XML represents the appropriate means to define such a data representation language. As a result WSDL (Web Services Description Language)—see www.w3c.org/TR/wsdl—was created providing the following constituents: • Types are used as core elements to build messages (XML Schema Notation). • Messages define packages exchanged within a single message transfer. Requests and responses represent separate messages. • Porttypes group messages to abstract operations. • Bindings map Porttypes to concrete protocols. • Ports denote the concrete communication addresses of services. • A Service comprises a collection of ports."* The preceding text excerpt clearly indicates that WSDL is used to describe the definition of the web services application control commands. Note that Step # indicates that the WSDL definition files are stored on the server.) (Page 74, Steps 2 and 3).

It would have been obvious to one skilled in the art at the time of Applicants invention to modify the teachings of Doshi with the teachings of Stal to include the definition of said application control commands is a web services description language (WSDL) file provided on the server with the motivation of providing a standard web service definition using WSDL.

As per Claims 4 and 16, Doshi fails to disclose said WSDL file includes said instructions which define said client-side web service.

Stal discloses said WSDL file includes said instructions which define said client-side web service (i.e. *"Step 2: Implementing an Interface Definition Language. In the next implementation step, an interface definition language is introduced to express structural and deployment information in an implementation- neutral fashion. From the specifications written in this language, tools will automatically generate the implementations of client-side and serverside proxies. Again, XML represents the appropriate means to define such a data representation language. As a result WSDL (Web Services Description Language)—see www.w3c.org/TR/wsdl—was created providing the following constituents: • Types are used as core elements to build messages (XML Schema Notation). • Messages define packages exchanged within a single message transfer. Requests and responses represent separate messages. • Porttypes group messages to abstract operations. • Bindings map Porttypes to concrete protocols. • Ports denote the concrete communication addresses of services. • A Service comprises a collection of ports."* The preceding text excerpt clearly indicates that WSDL is used to describe the definition of the web services application control commands. Note that Step # indicates that the WSDL definition files are stored on the server.) (Page 74, Steps 2 and 3).

It would have been obvious to one skilled in the art at the time of Applicants invention to modify the teachings of Doshi with the teachings of Stal to include said WSDL file includes said instructions which define said client-side web service with the motivation of providing a standard web service definition using WSDL.

As per Claims 5 and 17, Doshi fails to disclose said WSDL file includes instructions for the client to access said instructions which define said client-side web service over the network.

Stal discloses said WSDL file includes instructions for the client to access said instructions which define said client-side web service over the network (i.e. *"Step 2:*

Implementing an Interface Definition Language. In the next implementation step, an interface definition language is introduced to express structural and deployment information in an implementation- neutral fashion. From the specifications written in this language, tools will automatically generate the implementations of client-side and serverside proxies. Again, XML represents the appropriate means to define such a data representation language. As a result WSDL (Web Services Description Language)—see www.w3c.org/TR/wsdl—was created providing the following constituents: • Types are used as core elements to build messages (XML Schema Notation). • Messages define packages exchanged within a single message transfer. Requests and responses represent separate messages. • Porttypes group messages to abstract operations. • Bindings map Porttypes to concrete protocols. • Ports denote the concrete communication addresses of services. • A Service comprises a collection of ports." The preceding text excerpt clearly indicates that WSDL is used to describe the definition of the web services application control commands. Note that Step # indicates that the WSDL definition files are stored on the server.) (Page 74, Steps 2 and 3).

It would have been obvious to one skilled in the art at the time of Applicants invention to modify the teachings of Doshi with the teachings of Stal to include said WSDL file includes instructions for the client to access said instructions which define said client-side web service over the network with the motivation of providing a standard web service definition using WSDL.

As per Claims 6 and 18, Doshi discloses said application is a computer telephone integration (CTI) application in communication with a telephony network (Figure 1a), whereby the interface on the client enables a user of the client to control a device on said telephony network (i.e. Column 3, Lines 20-48, wherein the network gateway acts in response to commands by the client in order to control the application.) and the interface on the

Art Unit: 2165

server enables the CTI application to issue notifications of telephony network events or state changes relating to the device to the user of the client (i.e. Column 3, Lines 20-48, wherein the network gateway sends responses and status updates to the client. Also note Column 9, Lines 60-64 which indicate that notifications may be sent to the client which are independent of application control commands sent by the client (e.g. third party bridge notifications).).

As per Claims 7 and 19, Doshi discloses said data network is a packet switching network employing the transport communication protocol/internet protocol (TCP/IP) method of addressing packets from the client to the server and vice versa (i.e. *Column 6, Lines 12-20 which disclose that the connection I maintained over a standard internet connection.*).

As per Claims 8 and 20, Doshi discloses the data network is selected from a local area network, a wide area network, and the Internet (i.e. *Column 6, Lines 12-20 which disclose that the connection I maintained over a standard internet connection.*).

As per Claim 9, Doshi discloses said server is a web server running said application (i.e. Figure 1a).

As per Claim 10, Doshi discloses said server is a web server and said application runs on a computer in communication with said server (i.e. Figure 1a).

As per Claims 13, 21, and 22, Doshi discloses a computer program product, client (e.g. system), and method of remotely controlling an application from a client

computer across a data network, said application being under the local control of a server on the data network (i.e. Figure 1a), without maintaining a dedicated communications channel between the client and the server (i.e. *Column 6, Lines 12-20 which disclose that the connection is maintained over a standard internet connection.*), comprising the steps of: b) providing on the client an interface based on said definition which formulates said predefined commands and addresses said commands to the server-side service in response to inputs to the client (i.e. *Column 3, Lines 20-48, wherein the network gateway sends responses and status updates to the client. Also note Column 9, Lines 60-64 which indicate that notifications may be sent to the client which are independent of application control commands sent by the client (e.g. third party bridge notifications).*); and d) providing on the client said network addressable client-side service which monitors for receipt of said notification generation commands and which generates said notifications on the client in response thereto (i.e. *Column 3, Lines 20-48, wherein the network gateway sends responses and status updates to the client. Also note Column 9, Lines 60-64 which indicate that notifications may be sent to the client which are independent of application control commands sent by the client (e.g. third party bridge notifications).*); whereby the server-side service provides a network addressable control service to enable the client to control said application on the server (i.e. *Column 3, Lines 20-48, wherein the network gateway acts in response to commands by the client in order to control the application.*), and the client-side service provides a network addressable notification service to enable the server to notify the client of events occurring in the application independently of said application control commands received from the client (i.e. *Column 3, Lines 20-48, wherein the network gateway sends responses and status updates to the client. Also note Column 9, Lines 60-64*

which indicate that notifications may be sent to the client which are independent of application control commands sent by the client (e.g. third party bridge notifications).).

Doshi fails to disclose receiving from the server a definition of application control commands which cause a network addressable server-side service to execute one or more predefined procedures to control said application and receiving instructions over the data network which define a network addressable client-side service which executes one or more predefined procedures to generate notifications on the client in response to predefined notification commands received by the client over the network.

Stal discloses receiving from the server a definition of application control commands which cause a network addressable server-side service to execute one or more predefined procedures to control said application (i.e. *"Step 2: Implementing an Interface Definition Language. In the next implementation step, an interface definition language is introduced to express structural and deployment information in an implementation- neutral fashion. From the specifications written in this language, tools will automatically generate the implementations of client-side and serverside proxies. Again, XML represents the appropriate means to define such a data representation language. As a result WSDL (Web Services Description Language)—see www.w3c.org/TR/wSDL—was created providing the following constituents: • Types are used as core elements to build messages (XML Schema Notation). • Messages define packages exchanged within a single message transfer. Requests and responses represent separate messages. • Porttypes group messages to abstract operations. • Bindings map Porttypes to concrete protocols. • Ports denote the concrete communication addresses of services. • A Service comprises a collection of ports. Step 3: Implementing a Service Directory. Before a client can access a service, it must find the service. For this purpose, a central broker must be available that allows implementers to register their services as well as clients to locate these services. Again, XML denotes the core technology to store and retrieve service registrations. UDDI (Universal Discovery, Description, and Integration)—see*

www3.ibm.com/services/uddi/standard.html—provides all functionality expected from a service broker. In UDDI, servers use the Publishers API to register services as well as additional business information with the global repository (see Figure 3). Clients access the Inquiry API to browse the repository and retrieve service descriptions. SOAP is used as communication protocol in all interactions. The client obtains the WSDL description from the UDDI repository both dynamically or statically, generates a client-side proxy, and invokes the Web service." The preceding text excerpt clearly indicates that a service definition language (WSDL) is used to create a description of the application control commands and the interface, including responses to user input. Also this definition is available to be sent over the network from the server to the client. Note that the functionality of the commands described in the claim language is clearly disclosed by the excerpts taken from Doshi above.) (Page 74, Column 1, Paragraph 3, Column 2, Paragraph 1) and receiving instructions over the data network which define a network addressable client-side service which executes one or more predefined procedures to generate notifications on the client in response to predefined notification commands received by the client over the network (i.e. *"Step 2: Implementing an Interface Definition Language. In the next implementation step, an interface definition language is introduced to express structural and deployment information in an implementation- neutral fashion. From the specifications written in this language, tools will automatically generate the implementations of client-side and serverside proxies. Again, XML represents the appropriate means to define such a data representation language. As a result WSDL (Web Services Description Language)—see www.w3c.org/TR/wsdl—was created providing the following constituents:*

- Types are used as core elements to build messages (XML Schema Notation).*
- Messages define packages exchanged within a single message transfer. Requests and responses represent separate messages.*
- Porttypes group messages to abstract operations.*
- Bindings map Porttypes to concrete protocols.*
- Ports denote the concrete communication addresses of services.*
- A Service comprises a collection of ports.*

Step 3: Implementing a Service Directory. Before a client can access a service, it must find the service. For this purpose, a central broker must be available that allows implementers to register their services as well as clients to locate these services. Again, XML

denotes the core technology to store and retrieve service registrations. UDDI (Universal Discovery, Description, and Integration)—see www3.ibm.com/services/uddi/standard.html—provides all functionality expected from a service broker. In UDDI, servers use the Publishers API to register services as well as additional business information with the global repository (see Figure 3). Clients access the Inquiry API to browse the repository and retrieve service descriptions. SOAP is used as communication protocol in all interactions. The client obtains the WSDL description from the UDDI repository both dynamically or statically, generates a client-side proxy, and invokes the Web service." The preceding text excerpt clearly indicates that a service definition language (WSDL) is used to create a description of the application control commands and the interface, including responses to user input. Also this definition is available to be sent over the network from the server to the client. Note that the functionality of the commands described in the claim language is clearly disclosed by the excerpts taken from Doshi above.) (Page 74, Column 1, Paragraph 3, Column 2, Paragraph 1).

It would have been obvious to one skilled in the art at the time of Applicants invention to modify the teachings of Doshi with the teachings of Stal to include receiving from the server across the network said definition of application control commands, and receiving over the data network said instructions which define a network addressable client-side service with the motivation of providing a standard web service definition using WSDL.

As per Claim 23, Doshi discloses a system comprising a client and a server connected across a data network, the client and the server each being provided with a storage area for storing instructions to allow asynchronous interaction between the client and server, and the server having control of an application (i.e. Column 3, Lines 20-48, wherein the network gateway acts as the server to control the application (e.g. call session and voice

Art Unit: 2165

switch).), wherein: (I) the storage area of the server stores instructions which when executed are effective to cause the server to: a) provide a network addressable server-side service which executes one or more predefined procedures to control said application in response to predefined application control commands received by the server over the network (i.e. Column 3, Lines 20-48, wherein the network gateway acts as the server to control the application (e.g. call session and voice switch).); c) provide instructions accessible over the data network by the client which define a network addressable client-side service which executes one or more predefined procedures to generate notifications on the client in response to predefined notification commands received by the client over the network (i.e. Column 3, Lines 20-48, wherein the network gateway sends responses and status updates to the client. Also note Column 9, Lines 60-64 which indicate that notifications may be sent to the client which are independent of application control commands sent by the client (e.g. third party bridge notifications).); and d) provide on the server an interface between the application and the data network which is effective to issue one or more of said notification generation commands in response to notifications of events received from the application, said events occurring and said notification generation commands issuing independently of said application control commands received from the client (i.e. Column 3, Lines 20-48, wherein the network gateway sends responses and status updates to the client. Also note Column 9, Lines 60-64 which indicate that notifications may be sent to the client which are independent of application control commands sent by the client (e.g. third party bridge notifications).); and (II) the storage area of the client stores instructions which when executed are effective to cause the client to: b) provide on the client an interface based on said definition which formulates said predefined commands and addresses said commands

Art Unit: 2165

to the server-side service in response to inputs to the client (i.e. Column 3, Lines 20-48, wherein the network gateway sends responses and status updates to the client. Also note Column 9, Lines 60-64 which indicate that notifications may be sent to the client which are independent of application control commands sent by the client (e.g. third party bridge notifications).); and d) provide on the client said network addressable client-side service which monitors for receipt of said notification generation commands and which generates said notifications on the computer in response thereto (i.e. Column 3, Lines 20-48, wherein the network gateway sends responses and status updates to the client. Also note Column 9, Lines 60-64 which indicate that notifications may be sent to the client which are independent of application control commands sent by the client (e.g. third party bridge notifications).); whereby the server-side service provides a network addressable control service to enable the client to control said application on the computer (i.e. Column 3, Lines 20-48, wherein the network gateway acts in response to commands by the client in order to control the application.), and the client-side service provides a network addressable notification service to enable the server to notify the client of events occurring in the application, said events occurring and said notification generation commands issuing independently of said application control commands received from the client (i.e. Column 3, Lines 20-48, wherein the network gateway sends responses and status updates to the client. Also note Column 9, Lines 60-64 which indicate that notifications may be sent to the client which are independent of application control commands sent by the client (e.g. third party bridge notifications).).

Doshi fails to disclose providing a definition of said application control commands, said definition being accessible over the data network by a client to enable the client to provide an interface which formulates said predefined commands and

addresses said commands to the server-side service in response to inputs to the client, receiving from the server across the network said definition of application control commands, and receiving over the data network said instructions which define a network addressable client-side service.

Stal discloses providing a definition of said application control commands, said definition being accessible over the data network by a client to enable the client to provide an interface which formulates said predefined commands and addresses said commands to the server-side service in response to inputs to the client (i.e. "Step 2: *Implementing an Interface Definition Language*. In the next implementation step, an interface definition language is introduced to express structural and deployment information in an implementation- neutral fashion. From the specifications written in this language, tools will automatically generate the implementations of client-side and serverside proxies. Again, XML represents the appropriate means to define such a data representation language. As a result WSDL (Web Services Description Language)—see www.w3c.org/TR/wsdl—was created providing the following constituents: • Types are used as core elements to build messages (XML Schema Notation). • Messages define packages exchanged within a single message transfer. Requests and responses represent separate messages. • Porttypes group messages to abstract operations. • Bindings map Porttypes to concrete protocols. • Ports denote the concrete communication addresses of services. • A Service comprises a collection of ports. Step 3: *Implementing a Service Directory*. Before a client can access a service, it must find the service. For this purpose, a central broker must be available that allows implementers to register their services as well as clients to locate these services. Again, XML denotes the core technology to store and retrieve service registrations. UDDI (Universal Discovery, Description, and Integration)—see www3.ibm.com/services/uddi/standard.html—provides all functionality expected from a service broker. In UDDI, servers use the Publishers API to register services as well as additional business information with the global repository (see Figure 3). Clients access the Inquiry API to browse the repository and retrieve

service descriptions. SOAP is used as communication protocol in all interactions. The client obtains the WSDL description from the UDDI repository both dynamically or statically, generates a client-side proxy, and invokes the Web service." The preceding text excerpt clearly indicates that a service definition language (WSDL) is used to create a description of the application control commands and the interface, including responses to user input. Also this definition is available to be sent over the network from the server to the client.) (Page 74, Column1, Paragraph 3, Column 2, Paragraph 1), receiving from the server across the network said definition of application control commands (i.e. "Step 2: *Implementing an Interface Definition Language. In the next implementation step, an interface definition language is introduced to express structural and deployment information in an implementation- neutral fashion. From the specifications written in this language, tools will automatically generate the implementations of client-side and serverside proxies. Again, XML represents the appropriate means to define such a data representation language. As a result WSDL (Web Services Description Language)—see www.w3c.org/TR/wsdl—was created providing the following constituents:* • Types are used as core elements to build messages (XML Schema Notation). • Messages define packages exchanged within a single message transfer. Requests and responses represent separate messages. • Porttypes group messages to abstract operations. • Bindings map Porttypes to concrete protocols. • Ports denote the concrete communication addresses of services. • A Service comprises a collection of ports. Step 3: *Implementing a Service Directory. Before a client can access a service, it must find the service. For this purpose, a central broker must be available that allows implementers to register their services as well as clients to locate these services. Again, XML denotes the core technology to store and retrieve service registrations. UDDI (Universal Discovery, Description, and Integration)—see www3.ibm.com/services/uddi/standard.html—provides all functionality expected from a service broker. In UDDI, servers use the Publishers API to register services as well as additional business information with the global repository (see Figure 3). Clients access the Inquiry API to browse the repository and retrieve service descriptions. SOAP is used as communication protocol in all interactions. The client obtains the WSDL description from the UDDI repository both dynamically or statically, generates a client-side proxy,*

and invokes the Web service." The preceding text excerpt clearly indicates that a service definition language (WSDL) is used to create a description of the application control commands and the interface, including responses to user input. Also this definition is available to be sent over the network from the server to the client. Note that the functionality of the commands described in the claim language is clearly disclosed by the excerpts taken from Doshi above.) (Page 74, Column 1, Paragraph 3, Column 2, Paragraph 1), and receiving over the data network said instructions which define a network addressable client-side service (i.e. "Step 2: Implementing an Interface Definition Language. In the next implementation step, an interface definition language is introduced to express structural and deployment information in an implementation- neutral fashion. From the specifications written in this language, tools will automatically generate the implementations of client-side and serverside proxies. Again, XML represents the appropriate means to define such a data representation language. As a result WSDL (Web Services Description Language)—see www.w3c.org/TR/wsdl—was created providing the following constituents: • Types are used as core elements to build messages (XML Schema Notation). • Messages define packages exchanged within a single message transfer. Requests and responses represent separate messages. • Porttypes group messages to abstract operations. • Bindings map Porttypes to concrete protocols. • Ports denote the concrete communication addresses of services. • A Service comprises a collection of ports. Step 3: Implementing a Service Directory. Before a client can access a service, it must find the service. For this purpose, a central broker must be available that allows implementers to register their services as well as clients to locate these services. Again, XML denotes the core technology to store and retrieve service registrations. UDDI (Universal Discovery, Description, and Integration)—see www3.ibm.com/services/uddi/standard.html—provides all functionality expected from a service broker. In UDDI, servers use the Publishers API to register services as well as additional business information with the global repository (see Figure 3). Clients access the Inquiry API to browse the repository and retrieve service descriptions. SOAP is used as communication protocol in all interactions. The client obtains the WSDL description from the UDDI repository both dynamically or statically, generates a client-side proxy, and invokes the Web service." The preceding text excerpt clearly

indicates that a service definition language (WSDL) is used to create a description of the application control commands and the interface, including responses to user input. Also this definition is available to be sent over the network from the server to the client. Note that the functionality of the commands described in the claim language is clearly disclosed by the excerpts taken from Doshi above.) (Page 74, Column 1, Paragraph 3, Column 2, Paragraph 1).

It would have been obvious to one skilled in the art at the time of Applicants invention to modify the teachings of Doshi with the teachings of Stal to include providing a definition of said application control commands, said definition being accessible over the data network by a client to enable the client to provide an interface which formulates said predefined commands and addresses said commands to the server-side service in response to inputs to the client, receive from the server across the network said definition of application control commands, and receive over the data network said instructions which define a network addressable client-side service with the motivation of providing a standard web service definition using WSDL.

7. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the

shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Points of Contact

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Michael J. Hicks whose telephone number is (571) 272-2670. The examiner can normally be reached on Monday - Friday 10:00a - 7:00p.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Jeffrey Gaffin can be reached on (571) 272-4146. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Michael J Hicks

Application/Control Number: 10/741,516

Page 23

Art Unit: 2165

Art Unit 2165

Phone: (571) 272-2670

Fax: (571) 273-2670

Nervan Akel-Jalil
~~new date~~
Primary Examiner